International Journal of Robotics
Research and Development (IJRRD)
ISSN(P): 2250-1592; ISSN(E): 2278-9421
Vol. 7, Issue 1, Apr 2017, 1-6

DEVELOPMENT OF SELF BALANCING ROBOT WITH PID CONTROL

SHUBHANK SONDHIA, RANJITH PILLAI. R,

SHARAT S. HEGDE, SAGAR CHAKOLE & VATSAL VORA

Department of Mechatronics, SRM University, Kattankulathur, Kanchipuram, Tamil Nadu, India

ABSTRACT

The paper describes utilization of the classical problem of inverted pendulum and its application to realize self-balancing robot. It is a two wheel vehicle whose structural, mechanical and electronic components were assembled in such a manner that it produced an inherently unstable platform which is highly susceptible to tip off in one axis. The wheels of the robot were capable of independent rotation each driven by a high torque DC motor. Information about the angle of the device relative to the ground was obtained from a 6DOFIMU (Inertial Measuring Unit) sensor which comprises of an accelerometer and agyroscope. Information from the IMU was processed and filtered to obtain accurat evalues which were fed to the micro processor on board. The microprocessor processed the feedback using a PID algorithm to generate position control signals i.e. apply proportional force to the motors as given by the program logic in order to restore the balance or to bring it back to its original vertical position. Two wheeled balancing robots can be used in several applications with different perspectives such as intelligent gardeners and autonomous trolleys in hospitals, transportation in shopping malls, offices, airports, or an intelligent robot.

KEYWORDS: Inverted Pendulum, Sensor, Two Wheeled Vehicle, and PID Control

Received: Jan 21, 2017**; Accepted**: Mar 10, 2017; **Published**: Mar 17, 2017; **Paper Id.**: IJRRDAPR20171

Original Article

INTRODUCTION

Self-balancing robot has been enormously recognized which is based on electronic device and embedded control and being used as a human transporter in many area. The self-balancing BOT is based on the Inverted Pendulum model (IP). In order to balance at two-wheeled inverted pendulum robot it is necessary to have accurate information of the live tilt angle from using a measurement on it. Furthermore a controller needs to be implemented to compensate for said tilt (Sugie & Fujimoto 1998; Nuo & Hui 2008; Tomasicet al., 2013; Jin 2015, Pillai et al. 2016). An Inverted Pendulum is a classic control problem. The system is non-linear and unstable with one input signal and several output signals. It is virtually impossible to balance the pendulum in the inverted position without applying some external force to the system. A PID-controller can be incorporated to control the pendulum angle, since it is a Single-Input Single-Output (SISO) system. If the robot should be able to be controlled in regard to position, x, as well as the angle, it becomes a Multiple-Input Multiple-Output(MIMO) system and one PID-controller is not enough. Controlling multiple states is conveniently made through a state space controller.

Many researchers and engineers are working on inverted pendulum and its application to realize a self- balancing robot because of its unstable nature, high order multi-variables, nonlinear and strong coupling properties and mobility (Kim & Kwom 2011; Balasubramaniam et al. 2016). Self-balancing robot like the Segway (http://www.segway.com) has been absolutely recognized and used as a human transporter especially for policeman. Several companies are coming with specific design of robots. Recently, Lego Company designed as

LegWay robot in which the differential driven method has been brought in to design so the robot could move either on inclined plane or irregular surface by using remote control operation (http://www.teamhassenplug.org/robot/segway). It is an ideal object of mechatronics, which includes sensors, actuators and embedded control system. A small mobile inverted pendulum called JOE (Grasser et al. 2002) is controlled by a joystick, which can be kept in balance when ever moving and turning. A feedback control educational prototype TV (Lin and Tsai2009) was developed, which could move either on the level ground or on the sloped surface. An intelligent two-wheeled robot called Balance Bot (http://www.art-of- invention. com/robotics) was developed on which the obstacle function was implemented. A simple self-balancing robot with Lego was also constructed, which includes AVR controller and some sensors (Ferdinando et al. 2011). A low cost self-balancing vehicle has been developed in Brno University (Grepl et al.). The two-wheeled robot is the combination of inverted pendulum system and two wheeled mobile robot. This brings an interesting concept of creating a transporter for human. The inverted pendulum is not actuated by itself; it uses the gyroscopes and accelerometers to sense the inclination off the vertical axis. The controller generates torque signals to each motor for preventing system from falling down to the ground. Inverted pendulum is a control model in which the object can be controlled only by adding loads on it. This kind of novel challenge is implemented and such controller has attracted interests of many researchers in the field of agricultural and autonomous trolleys.

The Inverted Pendulum is amongst the most difficult systems to control in the field of control engineering. An Inverted Pendulum is a pendulum that has its centre of mass above its pivot point. It is often implemented with the pivot point mounted on a cart that can move horizontally and may be called a cart and pole system as shown in Figure 1. The aim of Inverted Pendulum (IP) was to balance an inverted pendulum vertically on a motor driven wagon. To achieve this, an appropriate controller was required.

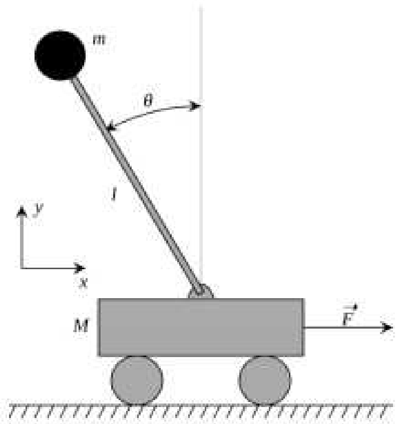

Figure 1: Inverted Pendulum

Therefore the reasons for selecting the Inverted Pendulum as the system were (i) It is the most easily available system (in most academia) for laboratory usage (ii) It is a non-linear system, which can be treated to be linear, without much error, for quite a wide range of variation. (iii) It provides a good practice for prospective control engineers.

SYSTEM DESCRIPTION

The bot consists of three platforms with the IMU on the topmost, microcontroller (which in our case was the ARM Cortex-M3SAM3X8E) board compatible with Arduino on the middle platform and the motor driver on the base

platform. On the upper part of the base platform batteries were installed and on the lower part of the base platform, the two high torque (35 kg-cm) motors of 100 rpm were clamped. The two wheels were mounted on the shafts of the high torque motors. The material used for the platform was acrylic board. The dimensions of each platform were 22.5 cm x 7cm with a gap of 5cms betweentwoconsecutiveplatforms.Screwsoflength6cmshavebeenusedto fix the layers and keep it intact. The whole bot balanced on two wheels having therequired grip which provided sufficient friction; pure mechanical balance of the bot was achieved (as the chances for the wheels to skid are large) (Figure 2).

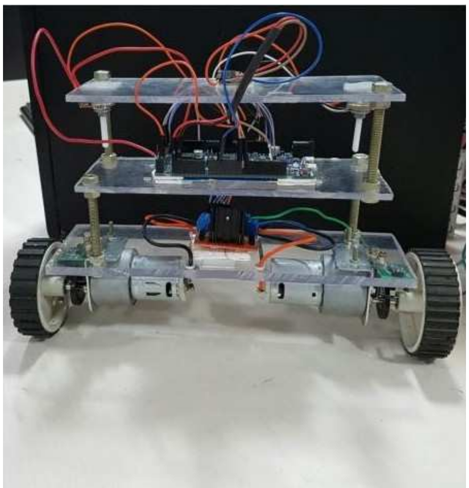

Figure 2: Mechanical Structure of Self Balancing Bot

ELECTRICAL SYSTEM DESIGN

Inertial Measuring Unit (IMU)

The IMU sensor contains a Micro Electro Mechanical System (MEMS) accelerometer and a MEMS gyro in a single chip. It is very accurate. It contains 16-bits analog to digital conversion hardware for each channel, therefore, it captures the x, y, and z channel at the same time. The sensor used the I2C-bus to interface with the microcontroller. The sensor sleep mode was disabled, and then the registers for the accelerometer and gyro were read. The sensor also contained a 1024 byte FIFO buffer. The sensor values are stored in the FIFO buffer and the buffer was read by the microcontroller. The FIFO buffer was used together with the interrupt signal. If the IMU places data in the FIFO buffer, it signals the microcontroller with the interrupt signal to apprise the microcontroller about the data in the FIFO buffer waiting to be read.

Sensor Fusion

The IMU had two sensors, an accelerometer and a gyroscope. The tri-axial accelerometer gave the components of acceleration (g) along its three axes. It was sensitive to noisy data. The gyroscope provided the angular velocity along its three axes. It was less sensitive than the accelerometer but its Output drifts from the actual value along with time. This was the reason sensor fusion becomes necessary as the values obtained from either of the senor is not completely reliable. The sensor had a ‘Digital Motion Processor’ (DMP), also called a "Digital Motion Processing Unit". This DMP can be programmed with firm ware and is able to do comple x calculations with the sensor values. The DMP can do fast calculations directly on the chip. This reduced the load for the microcontroller (like the Arduino). The values obtained from accelerometer and gyroscope was processed by DMP. It gave the yaw, pitch and roll of the vehicle. Here only value of the pitch is necessary as it gives the tilt value in the axis under consideration.

Algorithm– PID Control

The control algorithm that was used to maintain balance on the autonomous self balancing two wheel robot was the PID controller. The proportional, integral, and derivative (PID) controller is well known as a three term controller. The input to the controller was the error from the system. The *Kp, Ki,* and *Kd* were referred as the proportional, integral, and derivative constants (the three terms get multiplied by these constants, respectively). The closed loop control system is also referred to as a negative feedback system. The basic idea of an negative feedback system was that it measured the process output 'y' from a sensor. The measured process output gets subtracted from the reference set-point value to produce an error. The error was then fed into the PID controller, where the error got managed in three ways. The error was used on the PID controller to execute the proportional term, integral term for reduction of steady state errors, and the derivative term to handle over shoots. After the PID algorithm processed the error, the controller produced the control signal'u'. The PID control system then fed into the process under control. The process under PID control was the two wheeled robot. The PID control signal was try to drive the process to the desired reference set point value. In the case of the two wheel robot, the desired set-point value was the zero degree vertical position. The PID control algorithm can be modeled in a mathematical representation.

PID was used to calculate the "correction factor" *as shown below:*

$$\boldsymbol{correction = Kp * E + Ki * \int E(t)\, dt + Kd * [dE(t)/dt]} \quad \textbf{(4.1)}$$

Kp, ***Ki***,and ***Kd*** which are specific error constants, are set experimentally

E is the error signal constant

The integral term was simply the summation of all previous deviations and called this Integral as–"total error". The derivative was the difference between the current deviation and the previous deviation

Following was the code for evaluating the correction. These lines wererunin each iteration:

Correction=Kp*D + Ki*[(DT)*dt] + Kd[(D-D0)*dt] ***(4.2)***

D0=D

DT=D+DT

Where '**D**' is the deviation, '**D0**' is the previous deviation, '**DT**' is the total deviation or accumulated deviation and '**dt**' is the sampling time.

It was assumed that if only the first term had been used to calculate the correction, the robot would have reacted in the same way as in the classical line following algorithm. These cond term forced the robot to move towards the mean position faster. The third term resisted sudden change in deviation (Figure 3).

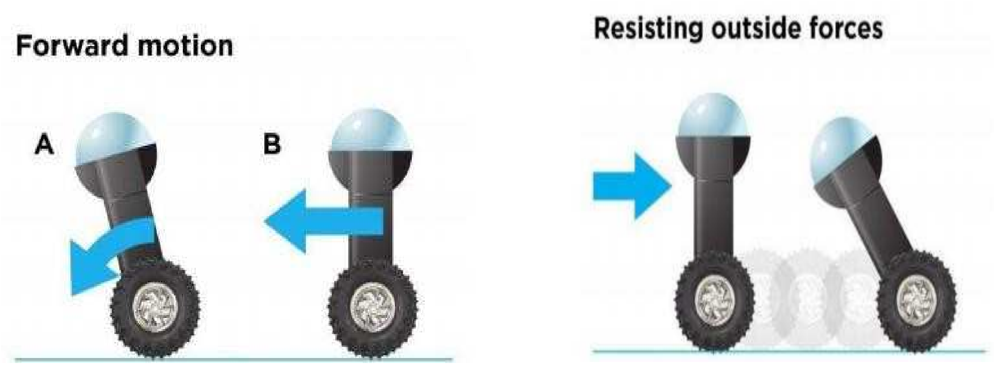

Figure 3: Balancing Technique of Bot

Overall System Design

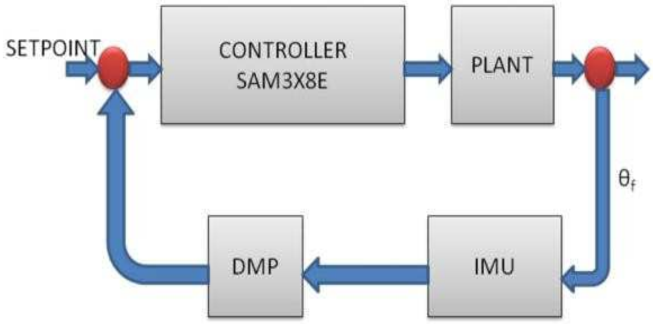

Figure 4: System Schematics

The IMU sensor mounted on top of the vehicle measured the acceleration and angular acceleration in three axes namely x, y and z. These values were processed by Digital Motion Processor (DMP) which transformed these values in to a more convenient set of variables i.e. yaw, pitch and roll. Here only pitch was necessary as it produced the value for tilt in the axis under consideration. This value was fed to the controller; which acted as feedback to the microcontroller. The micro controller processed the values obtained from DMP according to program specified algorithms. The controller compared the pitch value obtained from feedback with the pre-set value, if there was a deviation then the error value was sent to the PID controller which *via* its algorithm produced a proportional force to be applied on the motors in order to bring it to back to the original vertical position. The control signal was produced and sent to the motor controller L298N. The motor controller derived the motor at the specified speed, torque and direction (Figure 4).

CONCLUSIONS

The project utilized the classical problem of inverted pendulum and its application has been extended to realize self-balancing robot. The bot balances itself when pushed forward or backward using this principle. The requirements, functioning and connections of the components have been discussed in detail. The concept which was inspired from the functioning of the Segway can be further improved by using quadrature optical encoders for enhancing the precision of motor speed readings, which in turn would improve stability. Potentiometers can be used to tune the error constants of the PID control system. The mentioned features could not be incorporated due to the imposed time constraint but can be used as a modification to the existing system while designing a more efficient system in the future. Two wheeled balancing robots can be used in several applications with different perspectives such as intelligent gardeners in agricultural fields and autonomous trolleys in hospitals, shopping malls, offices, airports, healthcare applications, or an intelligent robot to guide

blind or disabled people.

ACKNOWLEDGEMENT

This paper entitled “Development of Self Balancing Bot with PID Control” is the work carried out at Motion Analysis Lab registered under the Department of Mechatronics at SRM University Kattankulathur, Tamil Nadu, India. We would like to thank the university valuable support and encouragement.

REFERENCES

1. *Pillai R., Sondhia S., Hegde S. S., Chakole S. & Vora V. (2016). Project report on self balancing robot using PID control. Motion Analysis Lab, Department of Mechatronics, SRM University Kattankulathur, Tamil Nadu, India*
2. *Ferdinando, H., Khoswanto H., & Tjokro, S. (2011). Design and evaluation of two-wheeled balancing robot chassis. IEEE International Conference on Communications, Computing and Control Applications, pp. 1-6.*
3. *Grasser, F., Arrigo, A. D., & ColombiS. (2002). JOE: Amobile, inverted pendulum.I EEE Transaction on Industrial Electronics 49, 107-114.*
4. *Grepl, R., Zouhar, F., Stepanek, J., & Horak, P. (2016). The development of a self-balancing vehicle: A platform for education in mechatronics. Faculty of Mechanical Engineering, Brno University of Technology. http://dsp.vscht.cz/konference_matlab/MATLAB11/prispevky/040_grepl.pdf(Accessed 16 February 2017).*
5. *http://www. art-of-invention. com/robotics*
6. *http://www. segway. com.*
7. *http://www. teamhassenplug. org/robot/legway.*
8. *Jin, D. (2015). Development of a stable control system for a segway. Available at: http://www.raysforexcellence.se/wp-content/uploads/2013/01/Dennis-Jin-Developmentof-a- stable-control-system-for-a-segway.pdf.[Accessed 02 May, 2015]*
9. *Kim, S. & Kwom S. (2011). SDRE based non linear optimal control of a two-wheeled balancing robot. Journal of Institute of Control, Robotics and Systems, 17, 1037-1043.*
10. *Sun, L., & Gan, J. (2010).Research in g of two-wheeled self-balancing robot based on LQR combined with PID. International Workshop on Intelligent systems and Applications, pp. 1-5.*
11. *Lin, S.C., & Tsai, C.C. (2009).Develop of a self-balancing human transportation vehicle for the teaching of feedback control. IEEE Transaction Education, 52(1)157-168.*
12. *MathWorks, (2012). Control Tutorials for MATLAB and SIMULINK, Inverted Pendulum. Availableat: http://ctms.engin.umich.edu/CTMS/index.php.exampleInvertedPendulum section-System Modeling. [Accessed 21 April 2015]*
13. *Nuo, J. & Hui, W. (2008). Nonlinear control of an inverted pendulum system based on sliding mode method. ACTA Analysis Functionalis Applicata, 9 (3), 234-237.*
14. *Sugie, T. & Fujimoto, K. (1998). Controller design for an inverted pendulum based on approximate linearization. International Journal of Robust and Nonlinear Control, 8(7), 585-597.*
15. *Takei, T., lmamura, R. & Yuta, S. (2009). Baggage transportation and navigation by a wheel inverted pendulum mobile robot. IEEE Transaction on Industrial Electronics, 56, 3985-3994.*
16. *Tomasic, T., Demetlika, A., & Crnekovic, M. (2012). Self-balance mobile robot tilter. Transactions of FAMENA, 36 (3), 23-32.*